

Boundary-scan design principles for efficient LSSD ASIC testing

by R. W. Bassett
M. E. Turner
J. H. Panner
P. S. Gillis
S. F. Oakland
D. W. Stout

A boundary-scan logic design method that depends only on level-sensitive scan design (LSSD) principles has been developed for IBM CMOS application-specific integrated circuit (ASIC) products. This technique permits comprehensive testing of LSSD ASICs with high signal input/output (I/O) pin counts, using relatively inexpensive reduced-pin-count automatic test equipment (ATE). This paper describes the LSSD logic structures required, the reduced-pin-count testing and burn-in processes used, and the ASIC product design decisions that must be made to establish a consistent boundary-scan implementation.

Introduction

A number of similarly structured logic design techniques have been developed during the last several years in response to various testing requirements associated with complex, very large-scale integrated (VLSI) digital components, boards, and systems. These techniques, known collectively as boundary-scan design methods, are

generally characterized by the inclusion of serially scannable shift register latches (SRLs) at the signal input/output (I/O) periphery of integrated circuit components (i.e., chips or modules).

Development of multivendor system test and maintenance interfaces is one VLSI requirement that has been addressed using boundary-scan design methods. These standard interfaces, also known as testability buses, are needed to satisfy the maintenance objectives of systems using boards from several different suppliers. An example of a testability bus design utilizing component-level boundary-scan as one of its features has been reported by Avra [1].

Production testing of assembled printed circuit boards is a second VLSI requirement that has been addressed using boundary-scan design methods. Board testing has become progressively more difficult because of escalating VLSI circuit densities, the widespread use of VLSI catalog components (e.g., microprocessors) from merchant vendors, and the increasing use of surface-mount packaging technology (SMT). Those factors have adversely affected both basic methods of board testing—functional testing and in-circuit testing.

Functional board test development has historically been an iterative and costly process. The use of high-density VLSI components has made this task even more difficult, since it has precipitated dramatic increases in average board design complexity. This increase in complexity has led to a practical need for simulation-based test development to obtain high-quality tests in a

©Copyright 1990 by International Business Machines Corporation. Copying in printed form for private use is permitted without payment of royalty provided that (1) each reproduction is done without alteration and (2) the *Journal* reference and IBM copyright notice are included on the first page. The title and abstract, but no other portions, of this paper may be copied or distributed royalty free without further permission by computer-based and other information-service systems. Permission to *republish* any other portion of this paper must be obtained from the Editor.

timely fashion. That approach has been hampered by the general lack of adequate simulation models for vendor VLSI catalog components. Even with such models, however, it is still difficult to develop an effective test capable of detecting all likely assembly-related defects (e.g., missing or shorted component pins) when the only permissible test access points are the board functional I/O connections.

For those reasons, functional board testing has ordinarily been either preceded or replaced by an in-circuit testing step. The in-circuit method requires the tester to have direct physical access to each signal node on the board. This is normally done using a bed-of-nails fixture to contact each node on the unpopulated back side of the board. In-circuit test equipment vendors usually also sell in-circuit test pattern libraries for VLSI catalog components. Those patterns can be combined to produce a test sufficient to verify the board assembly process, thus mitigating the need for simulation models.

However, the increasing use of SMT component packaging represents a significant obstacle to in-circuit testing. Surface-mount technology permits board designs to be much more densely populated with such components than would be possible with comparable pin-in-hole components. SMT allows very close component spacings and facilitates placement of components on both sides of a board. To permit in-circuit testing, though, SMT board designs must either sacrifice much of their density leverage (to maintain bed-of-nails access), or must require that complex and costly test fixtures be concurrently developed. Boundary-scan design methods offer a third alternative that provides board signal node access but avoids in-circuit-test fixturing requirements [2, 3]. These techniques replace the physical access points needed for in-circuit testing with equivalent logical access points (i.e., the boundary-scan latches) corresponding to the signal I/O pins of each component. The board testing applications of boundary-scan have led to development of the proposed IEEE standard 1149.1 [4], which is discussed further in the next part of this paper.

Reduction of the cost of logic component testing is a third VLSI requirement that can be addressed using boundary-scan design methods. One early example of a serial-scan design technique being used to assist integrated circuit testing on inexpensive, low-pin-count testers was reported by Zasio in 1983 [5]. Although that particular approach did not appear to completely satisfy all VLSI component testing needs (e.g., I/O circuit parametric testing), it did appear promising enough to warrant further evaluation.

An investigation into component testing cost reduction for IBM CMOS application-specific integrated circuit (ASIC) products was begun during the latter half of 1986.

This work focused on the attributes and testing needs of the high-density CMOS device families under development at that time. Those ASICs were next-generation successors to earlier IBM CMOS products [6, 7]. The earlier products required that the resulting ASIC component designs comply with the level-sensitive scan design (LSSD) rules [8, 9]. LSSD—which is needed to ensure high-quality, race-free, stuck-fault testing—was a requirement of the successor products as well.

Previous methods of LSSD ASIC testing have assumed that the automatic test equipment (ATE) would provide at least as many full-function pin channels as the corresponding device signal I/O counts. However, that assumption has unfavorable long-term economic implications, since device ATE costs scale directly with the supplied full-function pin count. For CMOS ASICs with signal I/O counts commonly in the 200–500-pin range, the previous methods have required 256-pin or 512-pin testers costing several million dollars each. Thus, the goal of this investigation was to define a reduced-cost testing process for ASICs with high signal I/O counts, by eliminating the need to use device testers equipped with an equivalent number of full-function pin channels. However, the reduced-cost approach was still required to be as complete and comprehensive as existing methods, and to deliver comparable or superior component quality levels.

LSSD permits all SRL-bounded circuits to be tested using only the subset of component signal I/O pins needed to perform LSSD scanning and clocking operations (i.e., the LSSD test-function I/Os). Thus, when SRLs are placed in close logical proximity to all other ASIC signal I/O pins (i.e., the data I/Os), it is possible to test virtually the entire component on ATE supplying only sufficient full-function channels to accommodate the LSSD test-function I/O pins. Furthermore, if the residual circuits not enclosed by SRLs have suitable complexity limits, comprehensive testing of those external circuits can also be done in a more cost-effective way. Such external circuit testing requires the ATE to provide either 1) a limited set of additional full-function pin channels (to be shared piecewise among the ASIC data I/O pins during external testing); or 2) a full set of additional low-cost, reduced-function pin channels. This boundary-scan-based testing process is called “reduced-pin-count testing.” It permits high-pin-count LSSD ASIC components to be efficiently tested on ATE equipped with 64 full-function pin channels [10].

Reduced-pin-count testing has become an important CMOS ASIC product development objective. However, it must also be carefully implemented in the context of other ASIC and system development considerations. Those considerations include boundary-scan design

impacts on system performance and ASIC circuit density, boundary-scan logic usability for other functional or testing tasks, ASIC testing and burn-in cost efficiency factors, and the additional development effort required to provide reduced-pin-count testing support in the ASIC circuit library and the associated computer-aided design (CAD) software.

The following sections of this paper describe the LSSD boundary-scan design rules and the reduced-pin-count testing process in more detail. The corollary reduced-pin-count burn-in process is also presented. Finally, the major ASIC product implementation issues are analyzed, focusing on the considerations stated above.

Boundary-scan design principles

The LSSD boundary-scan design method used for IBM CMOS ASICs is somewhat different from the proposed IEEE 1149.1 boundary-scan architecture [4]. It is instructive, therefore, not only to explain the LSSD boundary-scan approach, but also to compare it with the IEEE 1149.1 approach. This can best be accomplished by first reviewing the essential elements of the 1149.1 architecture. The LSSD boundary-scan method is then introduced and contrasted with the proposed IEEE standard.

• IEEE 1149.1 boundary-scan

The IEEE 1149.1 boundary-scan architecture requires that a standard test access port (TAP) be designed into each conforming component. The TAP is operated by means of a four-pin test-signal interface. The signal pins comprise a test clock (TCK), a test mode select (TMS), a serial test data input (TDI), and a serial test data output (TDO).

The basic elements of the 1149.1 architecture are illustrated in **Figure 1**. The test access port is controlled by an internal, synchronous finite-state machine consisting of sixteen states. Its prescribed behavior is governed by the values placed on the TMS input at the time of a rising edge transition on the TCK signal. The state machine is defined so that it can be initialized to a known reset state within six test clock cycles. (Note: IEEE 1149.1 also defines an optional test reset signal that permits immediate, asynchronous initialization at the expense of an extra test pin.) The TAP is required to contain a serially loadable instruction register and a one-bit scan bypass register. The 1149.1 method specifies that all component signal I/O pins (other than the test signal interface pins) must be directly connected to logically adjacent boundary-scan cells. Those cells must also be interconnected to form a single boundary-scan register operated under TAP control.

The proposed standard defines three mandatory instructions: 1) **BYPASS**—to permit board-level shift

register reconfiguration for more efficient scanning (using the bypass register); 2) **EXTEST**—to permit testing of board interconnect wiring (using the boundary-scan register); and 3) **SAMPLE**—to permit monitoring of signals entering and leaving a component during normal system operation (using the boundary-scan register). The basic 1149.1 boundary-scan cell design is depicted in **Figure 2**. Some variations on this theme are permitted, but all 1149.1-compatible cell designs must contain a multiplexor and latch combination in order to concurrently support both **EXTEST** and **SAMPLE** operation requirements. In **EXTEST** mode, the boundary-scan latches must be able to control all component output signals and monitor all component input signals. In **SAMPLE** mode, the boundary-scan latches must be able to simultaneously monitor all component functional input and output signals without impeding system functional signal flow.

The mandatory features of IEEE 1149.1 clearly indicate that it is primarily intended to facilitate board assembly verification. The **EXTEST** operation permits simple scan testing of the interconnect wiring between boundary-scan components. This can be done without requiring either in-circuit tester access to all pins on all components (which SMT has made difficult) or a detailed description of each component's internal functional logic (which is generally unavailable for vendor VLSI catalog components). By contrast, the proposed standard does not require any internal system logic testing operation at all. It does recommend that one be provided, however, and defines the rules for optional instructions (**RUNBIST** and **INTEST**) to permit internal logic testing either by invoking built-in self-test facilities or by using the boundary-scan register to apply a vendor-supplied scan test.

• LSSD boundary-scan

When a board is designed entirely with IBM LSSD components, a gate-level logic model of each component is available to the board designer. Thus, it is technically possible to automatically generate a test for the entire board, including its component interconnect wiring [11]. However, if the component quality levels are uniformly high, the dominant failures encountered during board testing are due to faulty component mounting or interconnection, and not to internal component faults. In such cases, the effort required to generate and apply tests for those internal faults is not justified. Board testing efficiency can be substantially improved, though, by generating tests only for faults associated with the component pins and the board wiring.

The test generation process can be simplified even further by minimizing the amount of board logic that must be analyzed in developing such tests. One way to do

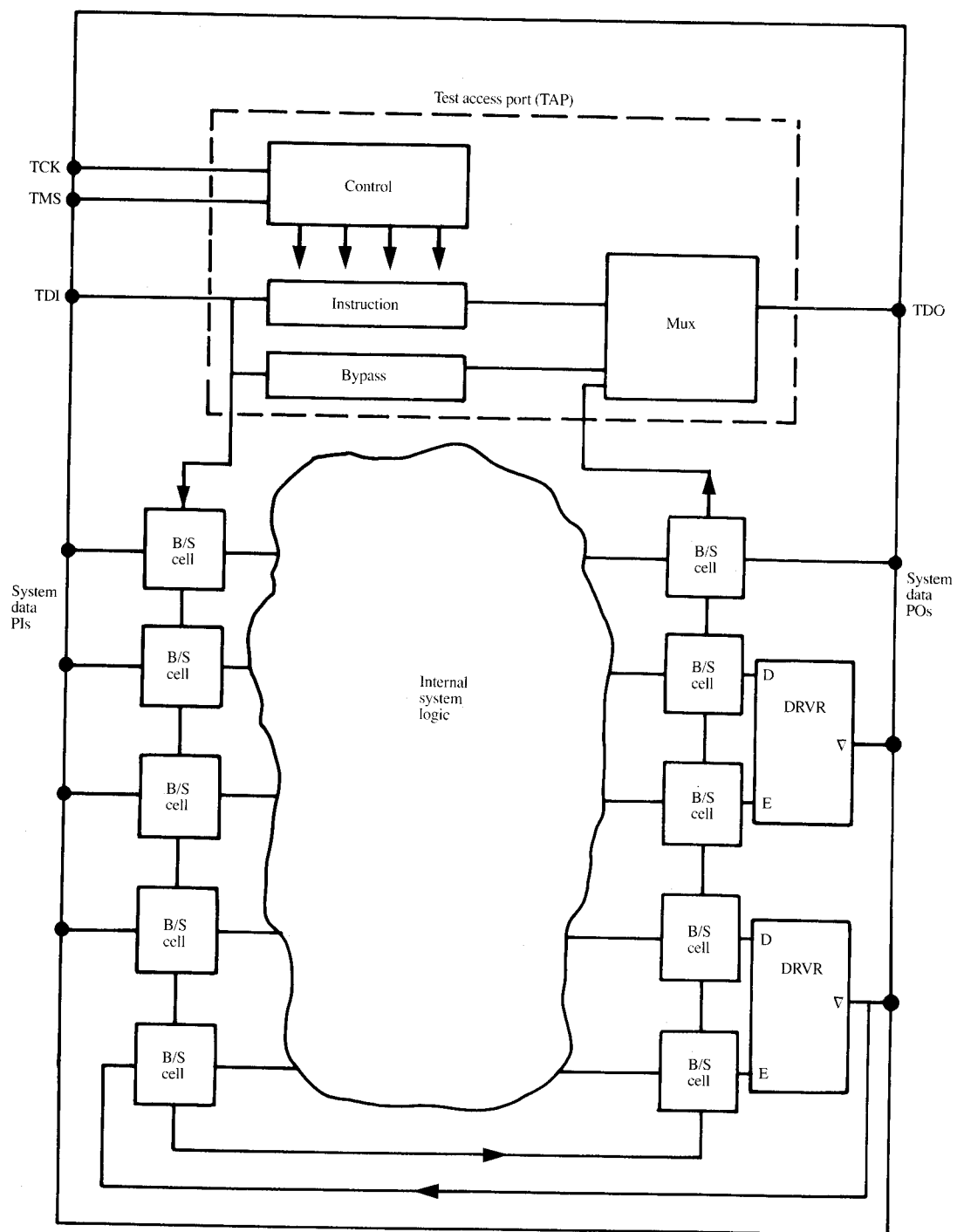


Figure 1

IEEE 1149.1 boundary-scan structure.

this would be to implement LSSD-compatible IEEE 1149.1 boundary-scan structures on each ASIC component (see Appendix A of [4]). Tests for the board interconnections could then be automatically generated and applied using the 1149.1 EXTEST protocol, which permits all internal-component functional logic to be ignored. The ASICs could continue to be treated as ordinary LSSD designs during component testing, however, since the logic added for 1149.1 compliance is indistinguishable from other functional logic for purposes of LSSD test generation.

A second way to simplify board test generation is to define a boundary-scan architecture based solely on the uniform application of LSSD principles. Two different methods of affiliating LSSD SRLs with component I/Os for board testing purposes were reported in 1982 and 1984 [12, 13]. Both approaches are formal means of ensuring LSSD logical partitionability at SRL boundaries that correspond closely to component signal I/O pin boundaries. The LSSD boundary-scan technique presented below is basically an extension of the full-chip partitioning aid (FCPA) method of DasGupta et al. [13]. Both FCPA and the following technique permit a logic network composed of one or more conforming components to be partitioned into two distinct, independently testable regions. The first region consists of the internal functional logic enclosed by the boundary SRLs (BSRLs) on each component. The second region consists of the component pins and any logic external to the BSRLs—principally the off-chip driver and receiver circuits attached to the I/O pins, and the interconnect wiring between board components.

The essential elements of LSSD boundary-scan component design are depicted in **Figure 3**. LSSD boundary-scan makes a formal distinction between pins required to perform specific testing functions and those having no such requirement. Thus, LSSD system clocks, scan clocks, scan gates, and scan data inputs (i.e., the inputs required for LSSD scanning and clocking operations) are categorized as test-function primary inputs (TFPIs). LSSD scan data outputs are categorized as test-function primary outputs (TFPOs), as are any other LSSD clock or control signals that may have to be distributed off-component for functional or testing purposes. All other input and output pins are classified as data primary inputs (data PIs) and data primary outputs (data POs), respectively. On the basis of this terminology, the following LSSD boundary-scan design rules apply:

Rule 1 There must be a TFPI sensitizing condition, consistent with the LSSD scan state and scan sequence, that makes all internal logic signals, and all embedded random-access memory (RAM) or read-only memory (ROM) arrays,

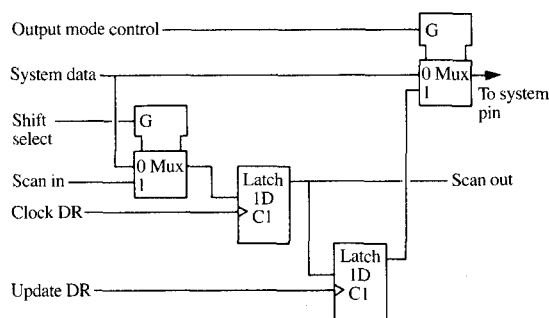


Figure 2

IEEE 1149.1 boundary-scan cell.

controllable and observable using only the TFPIs, TFPOs, and SRLs.

- Rule 2** There must be a TFPI sensitizing condition, consistent with the LSSD scan state and scan sequence, that makes all external logic signals, including all I/O pins (TFPIs, data PIs, TFPOs, and data POs), controllable and observable using only the BSRLs and the I/O pins themselves.
- Rule 3** All logic signals must be included in either the internal region, the external region, or both—and each possible value of a signal must be testable (i.e., simultaneously controllable and observable) under at least one of the two sensitizing conditions.

Rule 1 excludes data PIs and data POs from being used during internal testing operations. Rule 2 excludes any internal SRLs (i.e., those not designated as BSRLs) from being used during external testing operations. Rule 3, which is required for completeness, asserts that it must be possible to fully test all logic signals using the first two rules. These rules permit two basic BSRL arrangements for data PIs and POs. The data input structures are shown in **Figures 4** and **5**. The corresponding data output structures are depicted in **Figures 6** and **7**. Bidirectional data pins must use some combination of the data PI and PO configurations shown. It should be noted that the boundary-scan control inputs shown in **Figures 5** and **7** must be classified as TFPIs in order to satisfy the controllability and observability conditions of these rules.

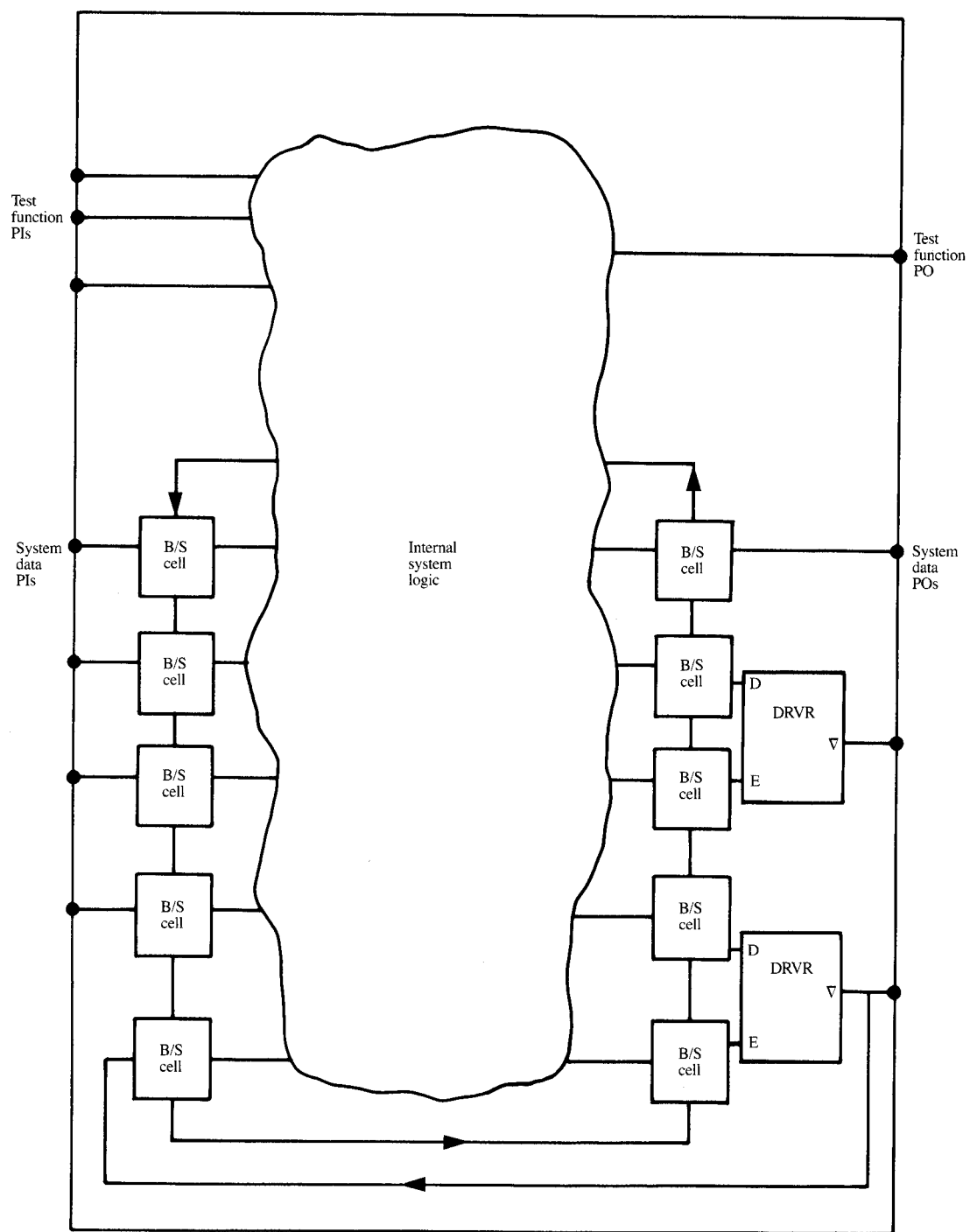


Figure 3

LSSD boundary-scan structure.



Figure 4

LSSD boundary-scan data input Example 1.

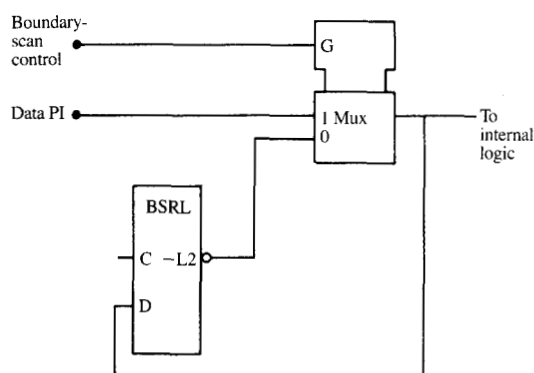


Figure 5

LSSD boundary-scan data input Example 2.

Figures 6 and 7 illustrate the further requirement that off-chip driver-enable signals fed by internal system logic must be intercepted by BSRL structures. Those figures also show a second driver-enable input that is controlled by a TFPI. This additional driver-inhibit control is included primarily for board testing. It provides a test-function input on each component that can be used by the board designer to prevent contention between three-state drivers on multisource board signal nodes during LSSD scanning operations.

Although Figures 4–7 do not show any combinational logic functions between data PIs, data POs, and boundary SRLs, such logic is not prohibited by the rules. However, Rule 1 does require that embedded RAMs or ROMs must be testable by the internal testing process. This restriction is needed to ensure a simple board-level interconnect wiring test.

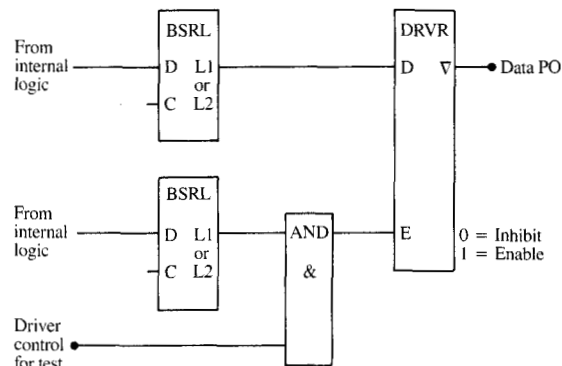


Figure 6

LSSD boundary-scan data output Example 1.

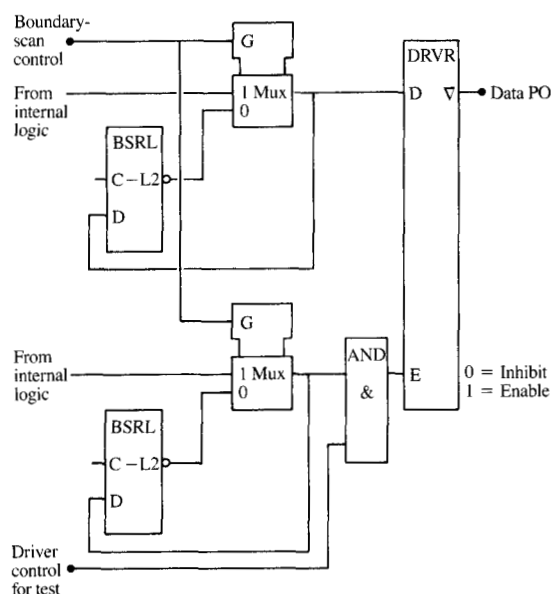


Figure 7

LSSD boundary-scan data output Example 2.

Rule 2 requires that all component pins, including the test-function I/Os, must be testable by the external testing process. This provision is also motivated by the need to

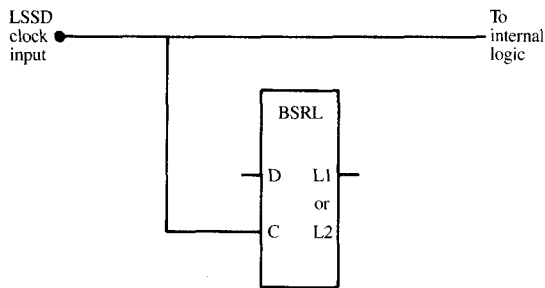


Figure 8

Rule 2 — LSSD clock TFPI.

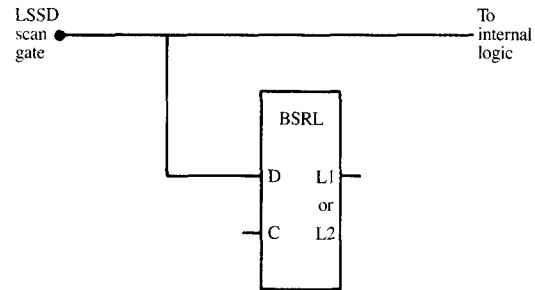


Figure 9

Rule 2 — LSSD scan gate TFPI.

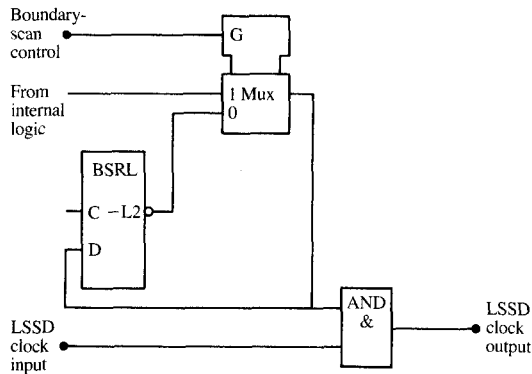


Figure 10

Rule 2 — LSSD clock TFPO.

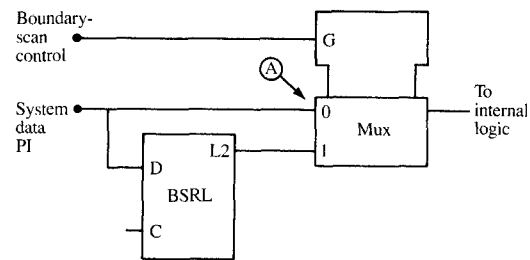


Figure 11

Incompletely testable boundary-scan cell.

ensure a simple and comprehensive board wiring test. Therefore, component designs must permit all TFPI signals to be sampled at BSRLs, and all TFPO signals to be generated using only BSRLs and TFPIs. These principles are illustrated in **Figures 8, 9, and 10**.

Rule 3—which requires that everything be testable under the conditions of Rules 1 and 2—might seem rather obvious, but it embodies a fairly subtle point. This rule disallows other incompletely testable multiplexor-SRL structures such as the one shown in **Figure 11**. In this figure, a stuck fault at node A cannot be tested under the provisions of either Rule 1 or Rule 2. During the internal testing process, node A cannot be driven to any

known value, since it is fed by a data PI. During external logic testing, node A can be controlled to an appropriate value, but the test outcome cannot be observed at the boundary SRL.

The preceding rules define a boundary-scan architecture that is both simpler and more flexible than the IEEE 1149.1 approach. LSSD boundary-scan does not require that a dedicated test access port be incorporated into each component to support boundary SRL scanning operations. A multiplexor-SRL boundary cell similar to the 1149.1 cell configuration is permitted, but a simpler approach using SRLs in series with system functional data paths is also supported.

Board-level testing support for internal component logic is only an optional feature of the IEEE 1149.1 standard. However, when a board design complies with the LSSD rules, the existence of a structure and method to test the internal component logic is guaranteed. Finally, the two boundary-scan methods are sufficiently similar to permit them to be used in tandem to test the interconnect wiring of a board containing both types of components; however, that discussion goes beyond the scope of this paper.

Reduced-pin-count testing

The LSSD boundary-scan rules provide a basic logical framework for reduced-pin-count component testing. Most, but not all, circuits on LSSD boundary-scan ASICs can be tested using ATE having only a sufficient number of full-function pin channels to manipulate the TFPI and TFPO pins. Such a test, though, is clearly inadequate to ensure shipment of uniformly high-quality components. Component stuck-fault test coverages in excess of 99% are essential to obtain shipped-product defect levels in the range of 1000 parts per million or less [14]. Thus, it is necessary for the ATE to support comprehensive testing of the external logic circuits as well.

The ATE resources required to support external testing are determined by the complexity of the logic circuits permitted to exist between the data PIs, data POs, and BSRLs. External logic complexity must be limited by ASIC product cell library design and the associated CAD process. These elements can be used to enforce a simple correspondence scheme between the data I/Os and boundary-scan cells similar to that required by FCPA [13]. This approach effectively allows only the off-chip driver and receiver circuits to be logically external to the boundary-scan cells.

ATE can support limited-complexity external logic testing in two basic ways. One method is to provide an additional limited set of full-function pin channels (in excess of those needed for TFPIs and TFPOs) to be shared among several data I/Os, whose affiliated circuits are then tested serially. However, from the perspective of overall testing cost, that approach can be quite complex. The cost advantages of ATE configured exclusively with a reduced number of full-function pin channels must be carefully balanced against the added fixturing costs and increased testing times (caused by sharing a small number of additional full-function pin channels among a large number of ASIC data I/O pins) of such an approach.

A second method is to equip the tester with only enough full-function pin channels (e.g., 64) to support LSSD TFPI and TFPO functions, and also with a second full set of low-cost, reduced-function pin channels (e.g., 448)—each providing a dedicated but limited capability,

adequate only for testing the data I/O circuits of boundary-scan components [10]. This approach results in the ATE-to-ASIC interface shown in Figure 12. As in conventional full-pin-count testing, there is still a direct correspondence between each ASIC signal I/O pin and a particular ATE channel. However, reduced-pin-count testing introduces the requirement that all component TFPI and TFPO pins must be mapped only onto full-function ATE channels. Data I/Os may be connected to either channel type.

This pin-to-channel mapping is determined by the wiring of the physical interface (e.g., wafer probe card) interposed between the ATE and the device to be tested. Practical manufacturing considerations dictate that a standard fixture be used for all ASIC components of the same product and package type. Thus, the fixture design for each package type establishes a 64-pin ASIC signal I/O subset that may be used for TFPI and TFPO functions. This design requirement must also be supported and checked by the ASIC CAD system.

The test-pattern generation process for LSSD boundary-scan components produces two distinct sets of test patterns. One set is generated for the internal testing operation (Rule 1), and a second set is generated for the external testing operation (Rule 2). As noted previously, the aggregate fault coverage of the two test sets must be greater than 99% to guarantee reasonable product quality objectives.

• Internal logic and embedded memory testing

The internal LSSD logic tests can be generated and applied in various ways. To satisfy component quality goals, the conventional approach has been to use a deterministic test-generation algorithm to develop a stored-pattern test set [15]. This test is then applied using conventional logic ATE, which must contain large amounts of stored-pattern memory for each full-function tester channel.

A second approach to internal logic testing is the weighted random-pattern (WRP) method of Waicukauski et al. [16, 17]. WRP also uses a deterministic algorithm to ensure comparably high stuck-fault coverage. These tests display superior properties (in comparison to conventional stored-pattern tests) for detection of certain classes of nonmodeled (i.e., nonstuck) faults as well. The WRP method produces approximately 10–50 times more tests than the conventional approach, however, so the tests must be applied using modified ATE. Each full-function ATE channel must contain special hardware to generate weighted pseudorandom-pattern inputs and collect product response signatures exactly as done by the WRP test-generation software. The WRP approach is particularly attractive for cost-effective internal logic testing of high-density CMOS ASICs [10].

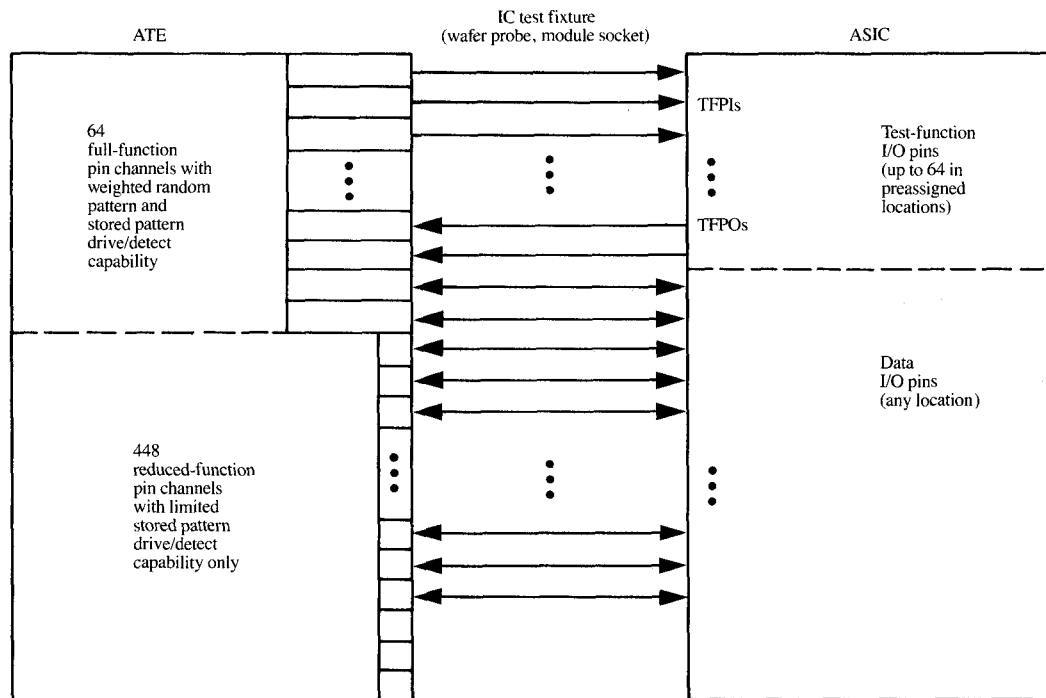


Figure 12

ATE-to-ASIC interface for reduced-pin-count testing.

Product quality requirements also dictate that deterministic methods be used to comprehensively test static RAM and ROM arrays that may be embedded in the ASIC internal logic. Specific functional sequences must be applied for effective testing, based on the particular functional and physical characteristics of each memory design [18]. In general, such prespecified tests can be applied by establishing a one-to-one correspondence between array inputs and LSSD-controllable points (PIs or SRLs), and between array outputs and LSSD-observable points (POs or SRLs) [19]. As the number and size of memories embedded on ASIC chips become large, however, this approach requires substantial amounts of ATE pattern storage. In addition, if SRLs are used as correspondence points, the time needed to apply the tests can be quite long because of the large number of scanning operations required (e.g., one LSSD scan per array read or write access).

To minimize the test time impact of scanning, previous IBM CMOS ASIC products have required that only component PIs and POs be used as correspondence points. In general, though, the I/O correspondence

method requires that a large number of component data PIs and POs be used to apply a large number of stored test patterns. This requirement is fundamentally inconsistent with the ATE cost-reduction objective of reduced-pin-count testing.

There are only two possible approaches, therefore, to reduced-pin-count memory testing. If the arrays are small and few in number, it may still be possible to test them economically using stored patterns by means of SRL-only correspondence. The other, more general alternative is to design the arrays to be self-testing [20]. Careful array self-test design can ensure high-quality memory testing while simultaneously avoiding the stored-pattern data volume and test time problems associated with the SRL-correspondence method. Some basic characteristics of LSSD-compatible array self-test design are subsequently described.

- *External logic testing*

LSSD tests for the external logic circuits are also generated using a deterministic algorithm. These tests are used to ensure not only the external logic function, but

also the electrical interface specifications of the off-chip driver and receiver circuits. To permit parametric testing of the driver and receiver circuits, the ATE reduced-function pin channels must be designed either with dedicated parametric measurement units (PMUs) or with shared access to PMUs elsewhere in the test system. These PMUs are used during external test application to verify that receiver switching threshold voltages and driver current capabilities satisfy their electrical specifications.

The external logic tests are applied using the stored-pattern approach. Receiver input thresholds are verified indirectly by applying the test pattern PI values using worst-case specified voltage levels (i.e., least positive up-level and most positive down-level). The test set includes patterns to detect the stuck faults on each PI, so all receiver threshold levels are verified when there are no functional miscompares during testing. Driver electrical characteristics are verified directly by taking appropriate analog measurements (i.e., source, sink, and leakage currents) under specified worst-case load conditions for each PO during external pattern application.

Both the external and internal tests are applied under carefully controlled power supply voltage and current conditions. The specified supply voltage tolerance limits (e.g., $\pm 10\%$ of the nominal value) are used during test application. At appropriate points during testing, power supply current measurements are taken as well. These parametric screening methods have proven to be an effective way to detect and eliminate marginal components that would otherwise cause intermittent system failures.

Another screening method, commonly known as burn-in, is also used to eliminate an additional class of marginal components—those that are initially functional even under parametric screening, but would soon fail during normal system use. The next section discusses the impact of boundary-scan design and reduced-pin-count testing constraints on the burn-in process.

Reduced-pin-count burn-in

Burn-in is the generic name for any process that exposes semiconductor products to elevated temperature and supply voltage conditions over an extended time interval (e.g., several hours). Several different burn-in methods can be used, and are usually classified according to their treatment of product I/O pins. These methods 1) apply a fixed, constant stimulus to the PIs, 2) apply a variable stimulus to the PIs, or 3) apply a variable stimulus to the PIs and measure the POs for expected fault-free response. The third category, which describes testing (under burn-in stress conditions), is called *in situ* burn-in.

The value of *in situ* burn-in as a means to ensure high-reliability components is well established. Testing during

burn-in stress has become a routine aspect of memory component manufacturing. *In situ* burn-in has been found to reduce early-life RAM failures (i.e., during the first 1000 power-on hours) by as much as a factor of 20 compared with unstressed components [21]. Because of their limited signal I/O pin counts, memory components have been testable using *in situ* burn-in systems equipped with no more than 64 tester pin channels [22]. Although some use of *in situ* burn-in for logic products (principally microprocessors) has been reported, these efforts have been hampered by the high tester pin counts required for typical logic components (and by the limited commercial availability of burn-in systems offering more than 64-pin capability). However, *in situ* burn-in is still needed to ensure logic reliability levels comparable to those of memory components, since the physical fabrication factors that influence reliability are virtually identical for logic and memory implemented with the same semiconductor process technology.

Burn-in improves component reliability by accelerating the exposure of latent physical defects that are sensitive to cumulative effects of operating temperature and voltage. One factor that determines the resulting reliability improvement is the percentage of all possible values on component signal nodes that are applied during burn-in. To ensure this factor, *in situ* burn-in requires that application of signal values be verified by tester measurement of expected component PO responses. These response measurements establish that the PI stimulus values needed for effective stress are actually being applied (i.e., that there has not been a burn-in equipment failure between the tester channels and the component PIs). PO response measurements also permit detection of components that exhibit recoverable fails (i.e., those that would retest as good under ambient temperature and nominal voltage conditions), which have been reported to comprise as much as 20% of the marginally reliable components that can be exposed by *in situ* burn-in.

In situ burn-in is performed on packaged components (modules), which are plugged into sockets on specially designed burn-in boards. Each board is populated with a collection of identical ASIC modules. These boards are then placed in a burn-in chamber, which provides the necessary temperature and voltage control facilities. During burn-in, a module test sequence is simultaneously applied to the PIs of all modules. Test output response from module POs is monitored for only one module at a time. The test sequence is continuously repeated through the duration of the burn-in, and all modules are monitored in rotation.

LSSD boundary-scan modules designed for reduced pin-count testing can receive a high-quality stress in this *in situ* burn-in environment. Almost all signal nodes on

such modules can be stress-tested using only the LSSD test-function I/O pin subset. For each module type, a standard 64-pin signal I/O subset that must include all TFPI and TFPO pins has been established. This information is also used to design standard, reusable burn-in boards for each module type. The corresponding test-function I/O pin locations for each module socket are wired in common on the burn-in board—with the exception of one or more fixed locations that must be wired individually, and reserved for driver-inhibit TFPIs. Independent control of driver-inhibit TFPIs is required to permit independent monitoring of the TFPOs on each module. Therefore, driver-inhibit TFPIs must occupy fixed pin positions within defined 64-pin test-function module I/O subsets, and the ASIC CAD system must enforce this additional burn-in requirement.

Under these conditions, a high-quality, cost-effective stress test can be provided for high-pin-count LSSD boundary-scan components, using a 64-pin *in situ* burn-in system and standard, reusable burn-in boards. All internal logic and array tests used for reduced-pin-count testing can also be applied during burn-in. A subset of the external logic test can also be used to stress—but not test—all driver circuits, and any receiver circuits connected to bidirectional data I/Os. Only circuits fed exclusively by data PIs cannot be stressed. By applying all internal tests and the external test subset as described, approximately 99% module stress coverage can be achieved.

ASIC product design considerations

The ASIC product development process includes the design of a cell library and a corresponding design automation system (CAD system). Several factors must be considered when integrating support for reduced-pin-count testing and burn-in into this process. These include ASIC testing and burn-in efficiency requirements, performance and density impacts of circuits added exclusively for testing, additional ASIC product development complexity, and overall product usability from a system design perspective. The ASIC cell library must contain all circuits needed to support reduced-pin-count testing. The CAD system must guarantee that each ASIC design satisfies the LSSD boundary-scan rules and other reduced-pin-count testing requirements. The following discussion focuses first on cell library design issues and then considers CAD system support issues in more detail.

• Boundary-scan and I/O cell design

The ASIC product development process must determine which of the boundary-scan structures shown in Figures 4–10 will be supported. A decision must be made regarding whether the product will use dedicated, test-

only BSRLs or will permit BSRLs to be used as functional system latches. A decision must also be made regarding whether the boundary SRLs will be located in the ASIC I/O cells or in the internal cell array. Differing product priorities can significantly influence these decisions, and two contrasting cases have been selected to illustrate this point. In the first case, the objectives are to achieve reasonably optimal test and burn-in efficiency, and to incur only modest additional cell library and CAD development costs. In the second case, the primary objective is to permit the maximum degree of system design flexibility that is still consistent with cost-effective reduced-pin-count testing.

The testing efficiency and development cost goals of the first case also indirectly imply that system design concerns have lower priority. Therefore, any performance or area penalties incurred due to boundary-scan will be minimized only to the extent permitted by ASIC product development cost and schedule constraints. An obvious way to achieve these goals is to implement a single standard-boundary SRL structure within the I/O cells—preferably using the simple series-latch configuration shown in Figures 4 and 6. In this case, the BSRLs are intended to be used only for testing, and to be transparent (i.e., flushed through) during normal system operation. Therefore, none of the typical design variations offered for functional SRL cells (extra data ports, clock gating, set/clear, etc.) need to be offered.

As a result, only one extra I/O cell design is needed for each driver and receiver circuit type, thus minimizing the number of unique I/O cells required. This approach sharply bounds the scope of additional development required to support reduced-pin-count testing, thereby satisfying the cost objective. The efficiency objective for testing and burn-in is satisfied by minimizing the number and complexity of circuits external to the BSRLs. Boundary SRL placement within the I/O cells permits circuit design optimization for a minimum number of transistors external to the scan-latch boundary. The performance and density impact of the BSRLs on the I/O cells can be reduced using ordinary circuit design techniques. By integrating the latch and I/O circuits, the functional path performance impact is limited to the delay of a single extra pass-gate. Circuit-switching performance is tuned for the functional data path, and not for BSRL operation during testing. The area penalty is also reduced, since the physical proximity of the latch and I/O circuits eliminates any requirement for intervening buffer stages.

In contrast, the second case assumes that system design flexibility is the most important objective. Therefore, increased ASIC product development costs may be incurred to eliminate arbitrary restrictions on the supported boundary-scan configurations, and to

minimize performance and circuit area impacts on system design. Marginal reductions in test and burn-in efficiency may also be permitted, although complete reduced-pin-count testing must still be guaranteed. From a system design perspective, it is preferable that the boundary SRLs be usable as functional system latches, or at least for system-level testing. High-performance systems typically require many component I/O signals to be functionally latched for timing reasons, thus creating a natural LSSD scan boundary. For I/O crossings that feed directly to or from system latches, additional nonfunctional BSRLs constitute unnecessary design overhead.

Therefore, it is desirable to permit system designers to construct boundary-scan structures personalized to the functional characteristics of each ASIC design by selecting an appropriate BSRL configuration for each I/O. This can best be accomplished by using SRLs in the ASIC internal cell library as boundary SRLs, instead of adding dedicated BSRLs to the I/O cells. No area or performance penalty is incurred when the BSRLs are used as functional system latches. Only one BSRL is needed when identical system-logic enabling conditions (e.g., for data or address buses) are used to control more than one component output driver (which also reduces the total number of SRLs that must be scanned during testing operations). A multiplexor-SRL combination circuit, configured as shown in Figures 5 and 7, is also provided in the ASIC cell library to help minimize boundary-scan design impact on data paths that are not functionally latched.

The system design must not be arbitrarily restricted in its use of LSSD system-clock gating for boundary SRLs, since such gating is generally required both for functional latch applications and for system-level testing. Two ASIC design features are needed to permit reduced-pin-count internal logic testing of system-clock gating for BSRLs fed by data PIs. The first feature is the inversion built into the multiplexor-SRL circuit shown in Figure 5, in which the BSRL L2-complement output is fed through the multiplexor and back to the BSRL L1 data input. The second feature is a receiver-inhibit TFPI signal that is connected to every data PI receiver cell, thus providing a way to force every data receiver output to a known, constant logic value during internal logic testing. These features permit the basic condition required for testing system-clock gating logic to be established for BSRLs fed by data PIs—placement of one logic value into the SRL by scanning, and placement of the opposite value on the SRL data input.

- *Embedded memory array design*

The ASIC product development process must also determine the design requirements for RAM and ROM

arrays to be included in the cell library. Depending on system requirements, some ASIC component designs may have no embedded memory arrays, while others may be dominated by the use of many or large arrays. The need to support a wide range of array usage presents a product development dilemma, as it creates a conflict between two important objectives—low-cost ASIC cell library development and high-quality, cost-effective testing and burn-in.

The low-cost library development objective favors a conventional approach to embedded memory design, and implies that testing must be done using stored patterns and SRL-only correspondence. This method is simplest from a circuit design perspective, but it can result in significant testing cost or quality problems for ASICs populated with large or numerous embedded arrays. These problems are due to the relatively large number of stored patterns needed to test even small memories, and to the large number of LSSD scanning cycles needed to apply them. ATE buffer storage capacity may not be sufficient to contain all patterns required for a high-quality test, and a severe test-time penalty is imposed if ATE buffer reloads must be done. This approach invariably results in poorer-quality testing and burn-in, since practical testing cost considerations ultimately limit the number of tests that can be applied.

In contrast, the testing and burn-in efficiency objective clearly points toward support of memory array self-test in the ASIC cell library. The simplest cell design approach is to incorporate self-test circuits into each embedded array library element. For a family of memory designs, a single self-test state machine can be designed as a modular logical and physical unit, and can be "grown" to fit each individual array configuration. This state machine generates algorithmic test patterns for the memory array, compares array outputs with expected values for all read operations, and sets a single SRL fail bit if any data miscompares are detected. The self-test circuits are fully LSSD-compatible, and are themselves tested during internal logic testing.

Careful array self-test design can greatly minimize test application times and ATE buffer storage requirements while also supporting high-quality testing. Only two LSSD scanning operations are needed, one to initialize the self-test state machine, and the other to observe the fail bits after completion of the tests. The self-test machine performs all testing operations between the two scans, and requires only that a simple, repetitive series of LSSD clock pulses be externally applied. The self-test machine is designed to enter a wait state when all tests have been applied, which permits all memory arrays on an ASIC to be tested in parallel. This technique requires only a minimal amount of ATE data buffer storage for the two LSSD scan operations and the self-test clocking

sequence. The self-test method permits practical use of more comprehensive memory test algorithms, thus contributing to improved component quality. For example, a self-test design has been developed that accesses each memory address 76 times, more than six times the number that would have been done for a comparable stored-pattern array test.

The performance impact of self-test design on functional memory operation can be minimized by proper implementation. The circuit design adds no delay to array read access time as measured from the clock input to data outputs. The set-up time of nonclock array control inputs is increased, but only by the delay of a single pass-gate placed in series with those inputs. The circuit area impact of self-test design on the size of individual memories can vary significantly, ranging from a large increase for small arrays (e.g., 20–25% for a 32×16 RAM), to a small increase for large arrays (e.g., 1–2% for an $8K \times 18$ RAM). However, small RAMs can also be implemented as scannable LSSD register arrays (which are tested as logic elements) to avoid excessive area penalties. Consequently, the total impact of array self-test design on CMOS ASIC area is relatively small (less than 5%), even if numerous arrays are used.

• CAD system support

The chief new requirement placed on the existing CAD system is to guarantee that each ASIC design satisfies reduced-pin-count testing constraints. This is done primarily by using physical design system features to guarantee restricted placement of designated test-function I/O cells, and to guarantee a correct boundary-scan structure for all data I/O cells.

The use of standard testing and burn-in fixtures for all ASICs of the same type requires definition of a standard set of 64 I/O pin locations for placement of TFPIs and TFPOs. The CAD system has rules that establish the legal placement locations for each library cell. The system uses these rules to perform cell placement, and later to check the design for compliance. These rules specify the standard 64-pin location set as the only permissible locations for library I/O cells designated for test-function I/O use.

For flexible boundary-scan implementations, the ASIC designer must specify the connections between BSRL cells and data I/O cells, consistent with the LSSD boundary-scan design rules. The CAD system must ultimately guarantee that this has been done properly. Special attributes (pin types) are associated with the pins of ASIC library cells, to allow or disallow connections to other pins. Here, specific pin types are associated with the data input and output pins of SRL cells (including multiplexor-SRLs) designated for BSRL use, and with pins of data I/O cells. The connection rules are as follows:

1. Each output pin of a data-receiver cell must be directly connected to one and only one data-input pin of an SRL or multiplexor-SRL cell. Some SRL cells may have more than one data-input pin and may be fed by more than one data-receiver cell.
2. Each data or system-enable input pin of a data-driver cell must be directly connected to one and only one data-output pin of an SRL or multiplexor-SRL cell. An SRL cell may feed more than one data-driver cell.

The CAD system checks the logic design for compliance with these rules, thereby preventing other cells from being connected between the I/O cells and the boundary SRL cells. ASIC cell library characteristics also constrain the complexity of features supported by the stated rules. In general, the amount of combinational logic contained within data I/O and SRL cells determines the amount of logic that must be tested during the external logic test.

A second CAD system requirement arises from the need to modify arbitrary connections between I/O cells during the physical design process. One example of such connections occurs for boundary-scan implementations using BSRLs located inside the I/O cells. In this case, the ASIC logic description must initially contain a logically correct, but functionally arbitrary, specification for LSSD clock and scan path connections between the I/O cells. Other examples include connections of the driver-inhibit lines used for board testing and module burn-in, and the receiver-inhibit line used for testing of BSRL clock-gating logic. These connections are also specified in a correct but arbitrary fashion prior to physical design.

As originally specified, these connections are generally incompatible with the preferred functional placement of the I/O cells, and may be completely unwirable for typical physically contiguous I/O socket arrangements. Therefore, the connections must be ignored during I/O cell placement, updated to reflect the actual physical ordering of the I/O, and then wired. The CAD system identifies these special connections using pin types, thereby avoiding any need for sophisticated path tracing. The pin-type information is used, in conjunction with a rule that describes the physical I/O socket ordering, to rebuild the affected portion of the ASIC logic specification. This permits automatic optimization of functionally arbitrary testing features without requiring system designer intervention in the ASIC physical design process.

Conclusions

A boundary-scan logic design method has been developed based on the uniform application of LSSD principles. This technique is an extension of the FCPA method of DasGupta et al. [13]. For LSSD ASICs, this boundary-scan method is simpler to use than the IEEE 1149.1

boundary-scan architecture, and it is also equally effective for purposes of board-level interconnect testing.

LSSD boundary-scan design is used to support a reduced-cost ASIC component testing process known as reduced-pin-count testing. The ASIC cell library design and CAD support must include appropriate provisions to ensure that all ASIC designs adhere to a rigorous boundary-scan implementation and satisfy all other specific reduced-pin-count testing and burn-in requirements. LSSD ASIC products can be developed using several different boundary-scan and reduced-pin-count testing support strategies. These approaches are characterized by differing ASIC cell library design and CAD support requirements, primarily for I/O circuits and embedded memory arrays. Selection of an appropriate strategy must be guided by conscious prioritization of the critical design and manufacturing objectives for each ASIC product family.

Acknowledgments

The authors acknowledge the contributions of many IBM colleagues to the development of the LSSD boundary-scan design method and the reduced-pin-count testing process described in this paper. While it is impractical to identify everyone who participated in this endeavor, we wish particularly to recognize J. Fiore, J. G. Petrovick, and L. K. Smith, Burlington; B. K. Koenemann and K. D. Wagner, Poughkeepsie; E. J. Orosz and T. J. Snethen, Endicott; P. K. Graham, Rochester; R. G. Walther, Austin; and T. W. Williams, Boulder.

References

1. L. Avra, "A VHSIC ETM-Bus-Compatible Test and Maintenance Interface," *Proceedings of the 1987 IEEE International Test Conference*, pp. 964-971.
2. C. Maunder and F. Beenker, "Boundary-Scan: A Framework for Structured Design-for-Test," *Proceedings of the 1987 IEEE International Test Conference*, pp. 714-723.
3. P. T. Wagner, "Interconnect Testing with Boundary Scan," *Proceedings of the 1987 IEEE International Test Conference*, pp. 52-57.
4. IEEE Computer Society Test Technology Technical Committee, "IEEE Standard Test Access Port and Boundary-Scan Architecture," *Draft Standard P1149.1-1989/D5*, Institute of Electrical and Electronics Engineers, Inc., New York, June 20, 1989.
5. J. J. Zasio, "Shifting Away from Probes for Wafer Test," *Proceedings of COMPCON Spring 1983*, pp. 395-398.
6. A. W. Aldridge, R. F. Keil, J. H. Panner, G. D. Pittman, and D. R. Thomas, "A 40K Equivalent Gate CMOS Standard Cell Chip," *Proceedings of the 1987 IEEE Custom Integrated Circuits Conference*, pp. 248-251.
7. R. Hornung, M. Bonneau, B. Waymel, J. Fiore, E. Gould, R. Piro, J. Martin, L. McAllister, and S. Tom, "A Versatile VLSI Design System for Combining Gate Array and Standard Cell Circuits on the Same Chip," *Proceedings of the 1987 IEEE Custom Integrated Circuits Conference*, pp. 245-247.
8. E. B. Eichelberger and T. W. Williams, "A Logic Design Structure for LSI Testability," *Proceedings of the 14th Design Automation Conference*, 1977, pp. 462-468.
9. S. DasGupta, P. Goel, R. G. Walther, and T. W. Williams, "A Variation of LSSD and Its Implications on Design and Test Pattern Generation in VLSI," *Proceedings of the 1982 IEEE International Test Conference*, pp. 63-66.
10. R. W. Bassett, B. J. Butkus, S. L. Dingle, M. R. Faucher, P. S. Gillis, J. H. Panner, J. G. Petrovick, and D. L. Wheeler, "Low Cost Testing of High Density Logic Components," *Proceedings of the 1989 IEEE International Test Conference*, pp. 550-557.
11. P. S. Bottorff, R. E. France, N. H. Garges, and E. J. Orosz, "Test Generation for Large Logic Networks," *Proceedings of the 14th Design Automation Conference*, 1977, pp. 479-485.
12. P. Goel and M. T. McMahon, "Electronic-Chip-In-Place Test," *Proceedings of the 19th Design Automation Conference*, 1982, pp. 482-488.
13. S. DasGupta, M. C. Graf, R. A. Rasmussen, R. G. Walther, and T. W. Williams, "Chip Partitioning Aid: A Design Technique for Partitionability and Testability in VLSI," *Proceedings of the 21st Design Automation Conference*, 1984, pp. 203-208.
14. E. J. McCluskey and F. Buelow, "IC Quality and Test Transparency," *Proceedings of the 1988 IEEE International Test Conference*, pp. 295-301.
15. P. Goel, "An Implicit Enumeration Algorithm to Generate Tests for Combinational Logic Circuits," *IEEE Trans. Computers* C-30, 215-222 (March 1981).
16. J. A. Waicukauski and E. Lindbloom, "Fault Detection Effectiveness of Weighted Random Patterns," *Proceedings of the 1988 IEEE International Test Conference*, pp. 245-255.
17. J. A. Waicukauski, E. Lindbloom, E. B. Eichelberger, and O. P. Forlenza, "A Method for Generating Weighted Random Test Patterns," *IBM J. Res. Develop.* 33, 149-161 (March 1989).
18. R. Dekker, F. Beenker, and L. Thijssen, "Fault Modeling and Test Algorithm Development for Static Random Access Memories," *Proceedings of the 1988 IEEE International Test Conference*, pp. 343-352.
19. E. B. Eichelberger, E. I. Muehldorf, T. W. Williams, and R. G. Walther, "A Logic Design Structure for Testing Internal Arrays," *Proceedings of the 3rd USA-Japan Computer Conference*, 1978, pp. 14-4-1-14-4-7.
20. R. Dekker, F. Beenker, and L. Thijssen, "A Realistic Self-Test Machine for Static Random Access Memories," *Proceedings of the 1988 IEEE International Test Conference*, pp. 353-361.
21. A. J. Wager, D. L. Thompson, and A. C. Forcier, "Implications of a Model for Optimum Burn-In," *Proceedings of the 21st IEEE Reliability Physics Symposium*, 1983, pp. 286-296.
22. C. Buck, "The Economic Benefits of Test During Burn-In: Real-World Experiences," *Proceedings of the 1987 IEEE International Test Conference*, pp. 1086-1093.

Received May 20, 1989; accepted for publication October 30, 1989

Robert W. Bassett IBM Systems Technology Division, Essex Junction, Vermont 05452. Mr. Bassett received the B.S.E.E. and M.S.E.E. degrees from Purdue University, West Lafayette, Indiana, in 1975 and 1977, respectively. He joined IBM in 1977 at the Rochester, Minnesota, development laboratory. While at Rochester, he worked in the areas of system diagnostic microcode development, logic chip and card testing support, and ASIC product development. From 1982 to 1986, Mr. Bassett worked at the IBM Burlington

laboratory in Essex Junction, where he was responsible for development of ASIC product test generation strategy. Since 1986, he has been a member of the VLSI design rules control department in the Engineering Design Systems (EDS) organization. His principal technical interests are in the areas of design-for-testability and test generation for VLSI components and systems. Mr. Bassett is a member of the Association for Computing Machinery and of the Institute of Electrical and Electronics Engineers.

Mark E. Turner *IBM General Technology Division, Essex Junction, Vermont 05452.* Mr. Turner received his B.S. degree in materials engineering from Rensselaer Polytechnic Institute in 1979, and his M.S. degree in electrical engineering from Lehigh University in 1981. He joined IBM at Essex Junction in 1981, working initially with a pilot line process characterization group. From 1982 to the present, Mr. Turner has had assignments dealing with various aspects of logic product development and testing, including CMOS fault analysis and boundary-scan design. During 1988 and 1989, he worked on timing analysis of VLSI chips and cards while on temporary assignment with a processor development group at the IBM laboratory in Rochester, Minnesota. He is currently a Senior Associate Engineer in the ASIC product development organization. Mr. Turner is a member of the Institute of Electrical and Electronics Engineers.

Jeannie H. Panner *IBM General Technology Division, Essex Junction, Vermont 05452.* Ms. Panner received her B.S. degree in mathematics from the State University of New York, Plattsburgh, in 1970. In 1974, she received an M.A. in mathematics from the University of Vermont, where she is now pursuing an M.S. in electrical engineering. Ms. Panner joined IBM in 1970 at Essex Junction. From 1970 to 1978, she worked on development of a wide range of manufacturing and scientific engineering applications. From 1978 to 1988, she was a project leader and system architect for ASIC logic chip CAD development. She is currently a Senior Engineer working on the design team for IBM's next-generation CAD system. Her technical interests are in the areas of CAD system architectures and frameworks for VLSI component and system design. Ms. Panner is a member of the Association for Computing Machinery and of the Institute of Electrical and Electronics Engineers.

Pamela S. Gillis *IBM General Technology Division, Essex Junction, Vermont 05452.* Ms. Gillis graduated from the University of California, Los Angeles, in 1971 with a B.A. in physics and astronomy. She received an M.S. in physics in 1973 and an M.A. in mathematics in 1974, both from the University of Colorado, Boulder. From 1974 to 1982, she worked at TRW Inc., in McClean, Virginia, doing engineering analysis studies in defense and energy conservation and leading several projects. In 1982, Ms. Gillis joined IBM at Essex Junction. From 1982 to the present, she has worked in the areas of test generation and testability analysis. Most recently, she has been a key contributor in the development of a new test methodology for IBM's next-generation VLSI chips and modules. Ms. Gillis is currently a Staff Engineer in the ASIC design automation organization.

Steven F. Oakland *IBM General Technology Division, Essex Junction, Vermont 05452.* Dr. Oakland received his B.S. and M.S. degrees from South Dakota State University in 1973 and 1984, respectively. He received his Ph.D. degree in electrical engineering and biomedical engineering from Iowa State University in 1980, joining IBM that same year at Rochester, Minnesota. While at

Rochester, he was involved with computer display terminal development. In 1983, Dr. Oakland joined the faculty of the Department of Computer Science, State University of New York, Plattsburgh, as a temporary assistant professor. Since 1984, he has been associated with the IBM Burlington development laboratory in Essex Junction, where he is currently an Advisory Engineer in the microsystem products organization. His principal activities and interests are in the areas of design-for-testability, design verification, and test generation.

Douglas W. Stout *IBM General Technology Division, Essex Junction, Vermont 05452.* Mr. Stout received his B.S. and M.S. degrees in electrical engineering from the University of Michigan, Ann Arbor, in 1981 and 1982, respectively. He joined IBM in 1983 at the Burlington development laboratory in Essex Junction, where he is currently a Staff Engineer in the ASIC product development organization. His principal activities and interests are in the areas of VLSI circuit design and logic product development. Mr. Stout is a member of the Institute of Electrical and Electronics Engineers.